Лабораторная работа №5

Модель эпидемии (SIR)

Плето Плето Мбамби

Содержание

# 1 Цель работы

Построить модель SIR в *xcos* и OpenModelica.

# 2 Задание

1. Реализовать модель SIR в в *xcos*;
2. Реализовать модель SIR с помощью блока Modelica в в *xcos*;
3. Реализовать модель SIR в OpenModelica;
4. Реализовать модель SIR с учётом процесса рождения / гибели особей в xcos (в том числе и с использованием блока Modelica), а также в OpenModelica;
5. Построить графики эпидемического порога при различных значениях параметров модели (в частности изменяя параметр );
6. Сделать анализ полученных графиков в зависимости от выбранных значений параметров модели.

# 3 Выполнение лабораторной работы

Задача о распространении эпидемии описывается системой дифференциальных уравнений:

где – скорость заражения, – скорость выздоровления.

## 3.1 Реализация модели в xcos

Зафиксируем начальные данные:

В меню Моделирование, Установить контекст зададим значения переменных и (рис. 1).

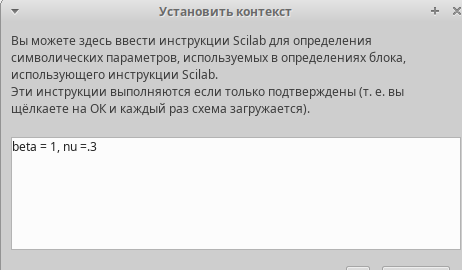


Figure : Рис. 1: Задание переменных окружения в xcos

Для реализации модели (рис. 2) потребуются следующие блоки xcos:

* CLOCK\_c – запуск часов модельного времени;
* CSCOPE – регистрирующее устройство для построения графика;
* TEXT\_f – задаёт текст примечаний;
* MUX – мультиплексер, позволяющий в данном случае вывести на графике сразу несколько кривых;
* INTEGRAL\_m – блок интегрирования;
* GAINBLK\_f – в данном случае позволяет задать значения коэффициентов и ;
* SUMMATION – блок суммирования;
* PROD\_f – поэлементное произведение двух векторов на входе блока.

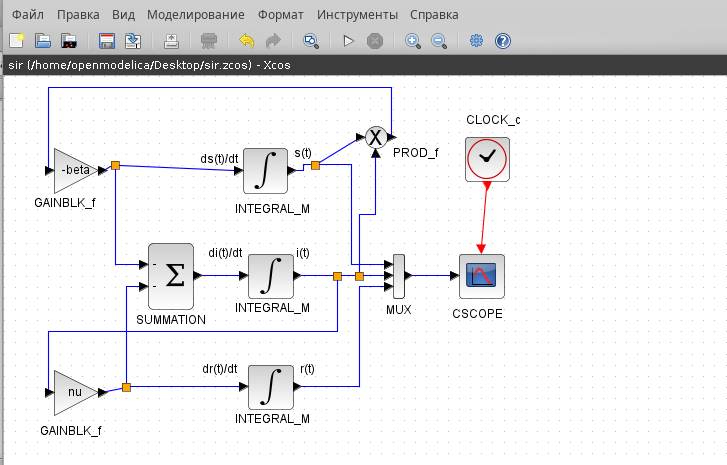


Figure : Рис. 2: Модель SIR в xcos

В параметрах верхнего и среднего блока интегрирования необходимо задать начальные значения и (рис. 3,4).

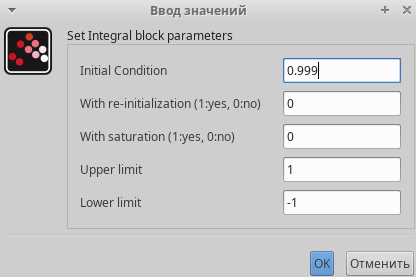


Figure : Рис. 3: Задание начальных значений в блоках интегрирования

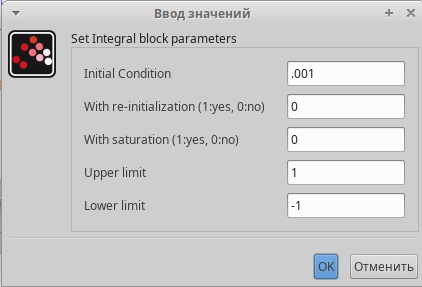


Figure : Рис. 4: Задание начальных значений в блоках интегрирования

В меню Моделирование, Установка зададим конечное время интегрирования, равным времени моделирования, в данном случае 30 (рис. 5).

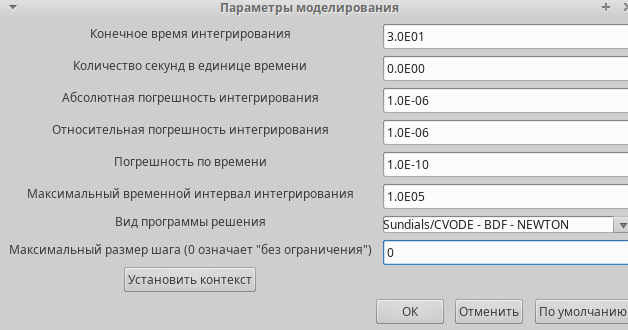


Figure : Рис. 5: Задание конечного времени интегрирования в xcos

Результат моделирования представлен на рис. 6, где черной линией обозначен график (динамика численности уязвимых к болезни особей), красная линия определяет — динамику численности выздоровевших особей, наконец, зеленая линия определяет — динамику численности заражённых особей. Пересечение трёх линий определяет порог эпидемии.

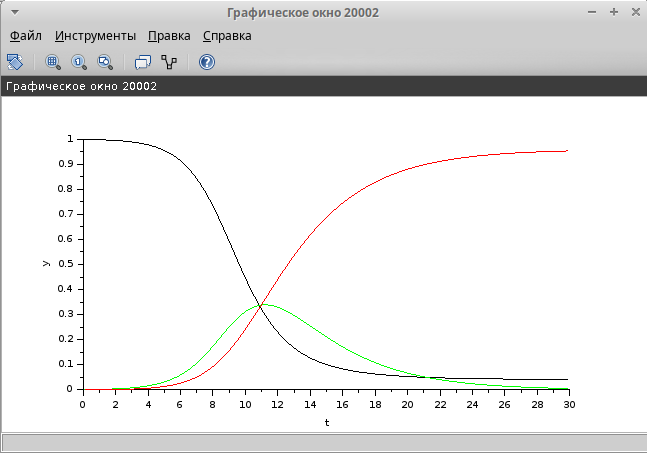


Figure : Рис. 6: Эпидемический порог модели SIR при

## 3.2 Реализация модели с помощью блока Modelica в xcos

Готовая модель SIR представлена на рис. 7.

Для реализации модели SIR с помощью языка Modelica помимо блоков CLOCK\_c, CSCOPE, TEXT\_f и MUX требуются блоки CONST\_m — задаёт константу; MBLOCK (Modelica generic) — блок реализации кода на языке Modelica. Задаём значения переменных и (рис. 1).

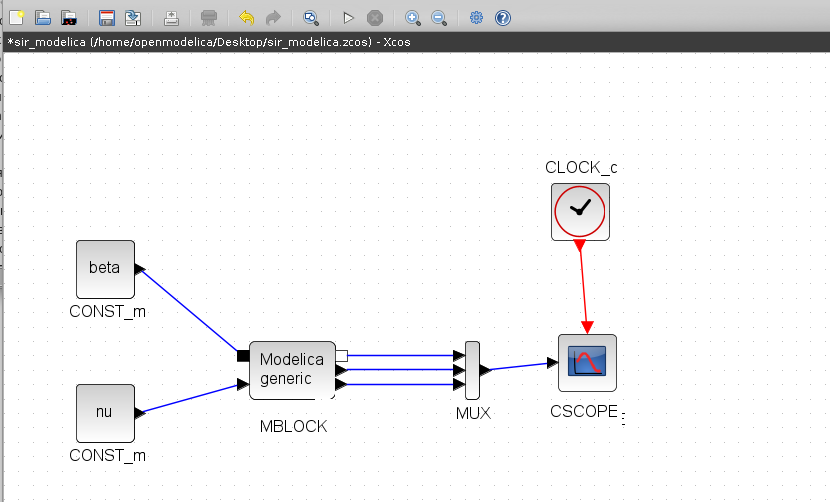


Figure : Рис. 7: Модель SIR в xcos с применением блока Modelica

Параметры блока Modelica представлены на рис. 8,9. Переменные на входе (“beta”, “nu”) и выходе (“s”, “i”, “r”) блока заданы как внешние (“E”).

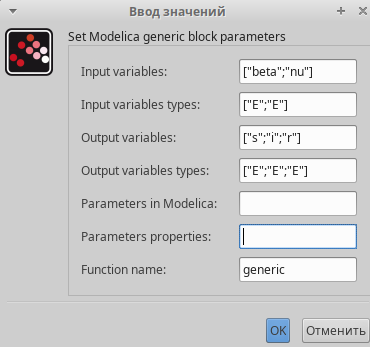


Figure : Рис. 8: Параметры блока Modelica для модели SIR

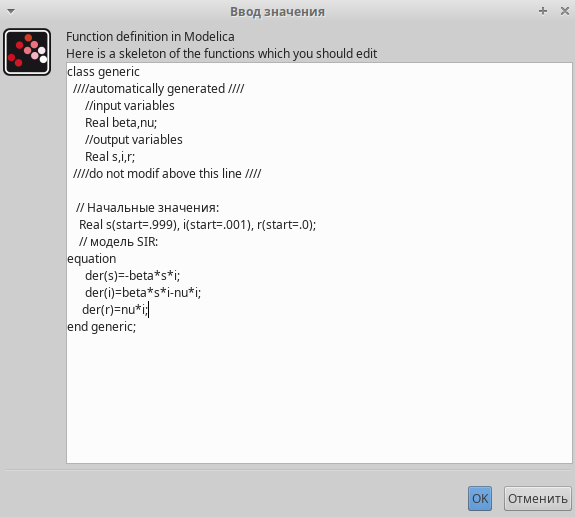


Figure : Рис. 9: Параметры блока Modelica для модели SIR

В результате получаем график (рис. 10), построенный с помощью блока Modelica идентичный графику (рис. 6), построенному без них.

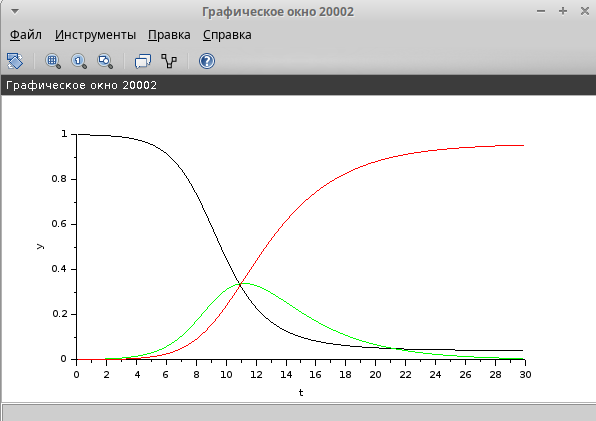


Figure : Рис. 10: Эпидемический порог модели SIR при

## 3.3 Упражнение

В качестве упражнения нам надо построить модель SIR на OpenModelica. Синтаксис почти такой же как и на Modelica. Нужно задать параметры, начальные значения и систему дифференциальных уравнений.

parameter Real I\_0 = 0.001;  
 parameter Real R\_0 = 0;  
 parameter Real S\_0 = 0.999;  
 parameter Real beta = 1;  
 parameter Real nu = 0.3;  
 parameter Real mu = 0.5;  
   
 Real s(start=S\_0);  
 Real i(start=I\_0);  
 Real r(start=R\_0);  
   
equation  
 der(s)=-beta\*s\*i;  
 der(i)=beta\*s\*i-nu\*i;  
 der(r)=nu\*i;

Теперь выполним симуляции, задав конечное время 30 с (рис. 11).

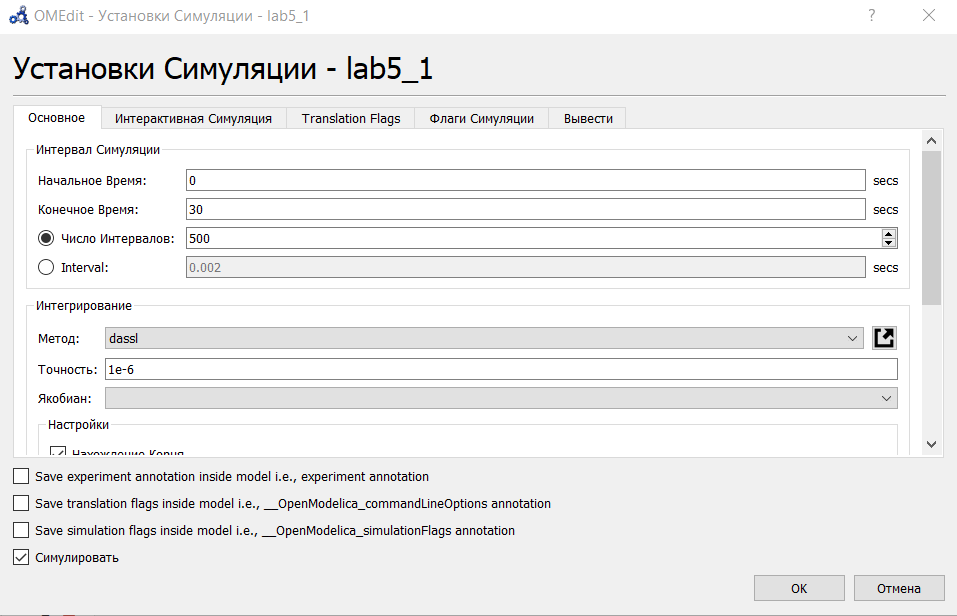


Figure : Рис. 11: Установка симуляции в OpenModelica

В результате получаем следующий график (рис. 12). Он идентичен предыдущим графикам выполненным в *xcos*.

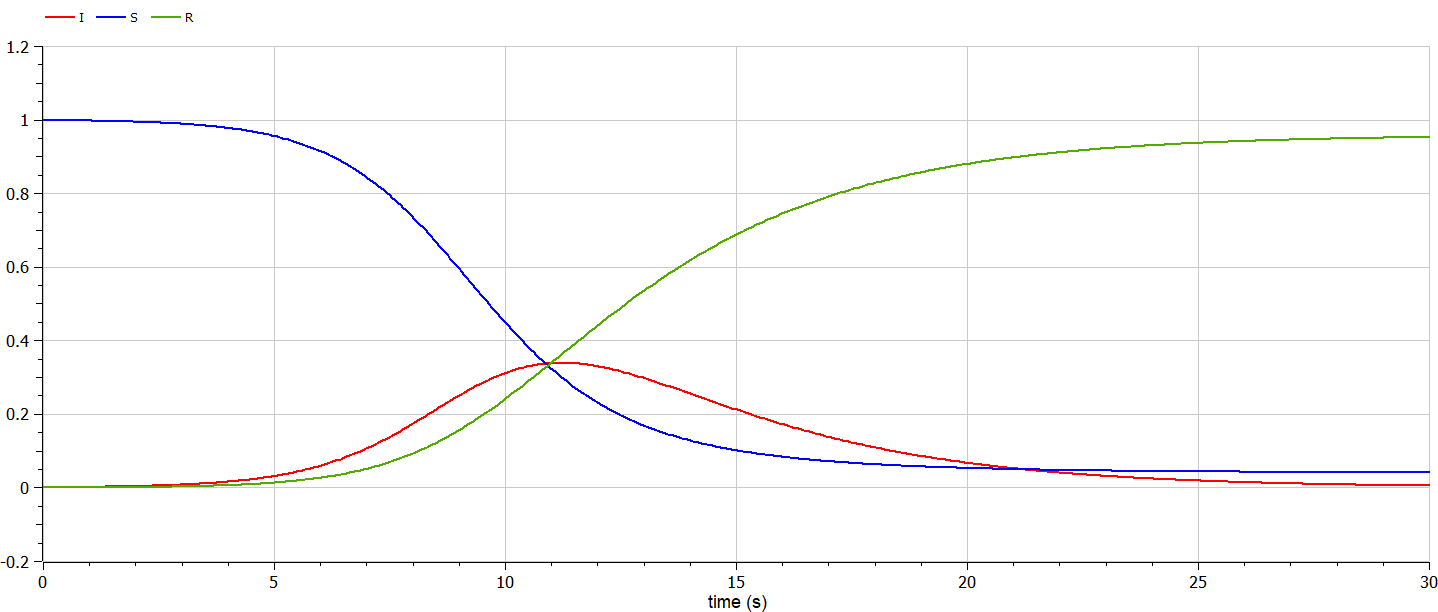


Figure : Рис. 12: Эпидемический порог модели SIR при

## 3.4 Задание для самостоятельного выполнения

Предположим, что в модели SIR учитываются демографические процессы, в частности, что смертность в популяции полностью уравновешивает рождаемость, а все рожденные индивидуумы появляются на свет абсолютно здоровыми. Тогда получим следующую систему уравнений:

где — константа, которая равна коэффициенту смертности и рождаемости.

Реализуем эту модель в *xcos*. Тут нам понадобятся три блока суммирования и 4 блока констант (добавляется константа ).

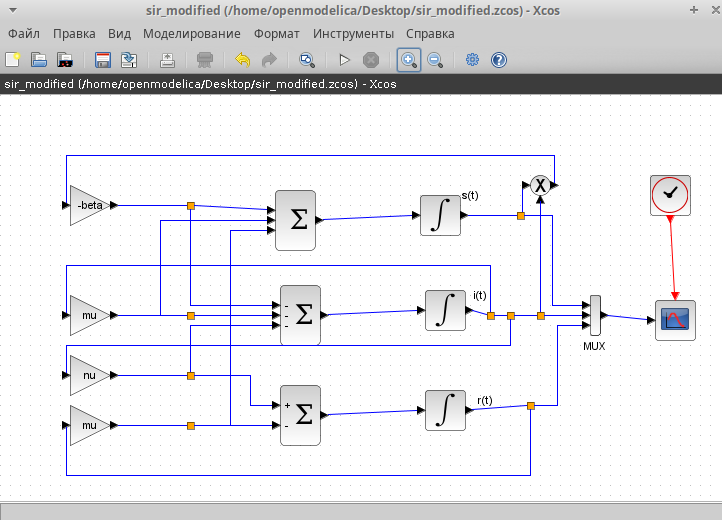


Figure : Рис. 13: Модель SIR с учетом демографических процессов в xcos

В результате получаем следующий график (рис. 14).

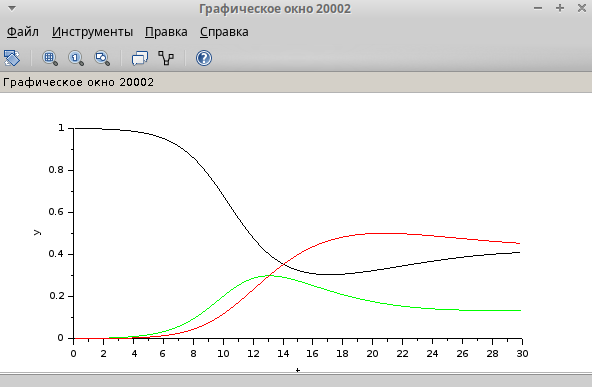


Figure : Рис. 14: График модели SIR с учетом демографических процессов

Теперь реализуем модель SIR с учетом демографических процессов в *xcos* с помощью блоков Modelica (рис. 15).

Рис. 15: Модель SIR с учетом демографических процессов в xcos с применением блока Modelica

Figure : Рис. 15: Модель SIR с учетом демографических процессов в xcos с применением блока Modelica

Параметры блока Modelica представлены на рис. 16,17. Переменные на входе (“beta”, “nu”, “mu” ) и выходе (“s”, “i”, “r”) блока заданы как внешние (“E”).

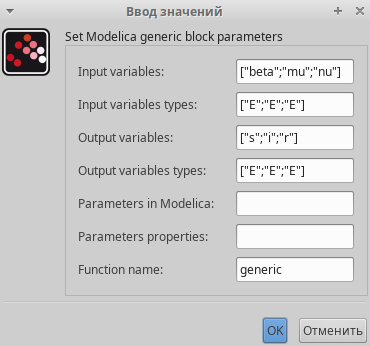


Figure : Рис. 16: Параметры блока Modelica для модели SIR с учетом демографических процессов

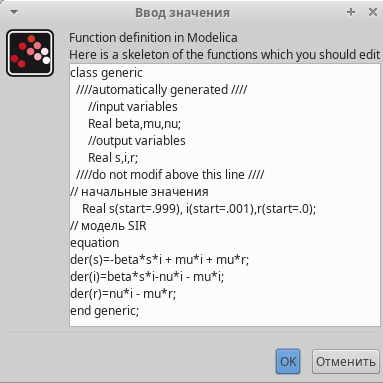


Figure : Рис. 17: Параметры блока Modelica для модели SIR с учетом демографических процессов

В результате получаем следующий график (рис. 18).

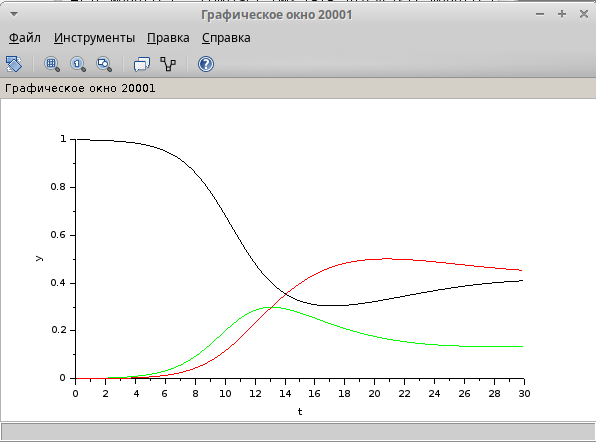


Figure : Рис. 18: График модели SIR с учетом демографических процессов

Реализуем модель SIR с учетом демографических процессов на OpenModelica.

parameter Real I\_0 = 0.001;  
 parameter Real R\_0 = 0;  
 parameter Real S\_0 = 0.999;  
 parameter Real N = 1;  
 parameter Real beta = 1;  
 parameter Real nu = 0.3;  
 parameter Real mu = 0.5;  
   
 Real s(start=S\_0);  
 Real i(start=I\_0);  
 Real r(start=R\_0);  
   
equation  
 der(s)=-beta\*s\*i + mu\*i + mu\*r;  
 der(i)=beta\*s\*i-nu\*i - mu\*i;  
 der(r)=nu\*i - mu\*r;

Выполнив симуляцию, получим следующий график (рис. 19).

Рис. 19: График модели SIR с учетом демографических процессов

Figure : Рис. 19: График модели SIR с учетом демографических процессов

Теперь построим графики при разных значениях параметров.

1. ,

Рис. 20: График модели SIR с учетом демографических процессов

Figure : Рис. 20: График модели SIR с учетом демографических процессов

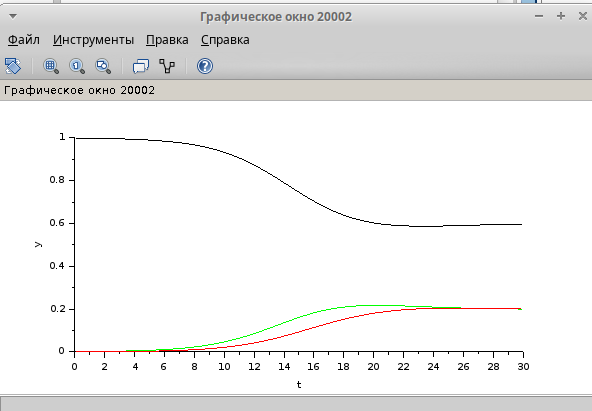


Figure : Рис. 21: График модели SIR с учетом демографических процессов

Рис. 22: График модели SIR с учетом демографических процессов

Figure : Рис. 22: График модели SIR с учетом демографических процессов

1. ,

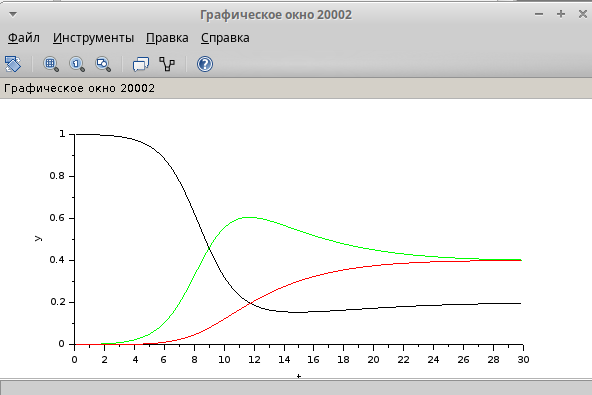


Figure : Рис. 23: График модели SIR с учетом демографических процессов



Figure : Рис. 24: График модели SIR с учетом демографических процессов

1. ,

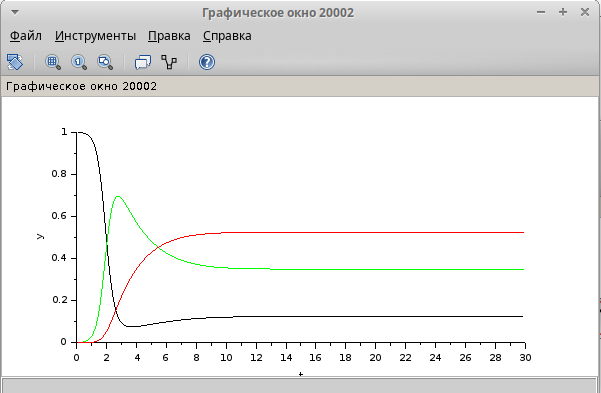


Figure : Рис. 25: График модели SIR с учетом демографических процессов

Исходя из анализа графиков, можно сделать вывод, что чем выше значение любого из параметров, тем быстрее система достигает стационарного состояния. При высоком коэффициенте заражения система быстро проходит через пик развития эпидемии и достигает стационарного состояния.

# 4 Выводы

В процессе выполнения данной лабораторной работы была построена модель SIR в *xcos* и OpenModelica.